



[> home](#) [> about](#) [> feedback](#) [> login](#)

US Patent & Trademark Office

## Search Results

Search Results for: [(object oriented) AND (chat)<AND>(meta\_published\_date <= 09-01-1999 )]

Found 102 of 104,445 searched. → Rerun within the Portal

Search within Results

---



[> Advanced Search](#) [> Search Help/Tips](#)

---

**Sort by:** Title Publication Publication Date Score Binder

---

**Results 1 - 20 of 102** short listing



1 2 3 4 5 6

- 1** Lightweight object-oriented shared variables for distributed applications on the Internet 92%

Jacob Harris , Vivek Sarkar  
ACM SIGPLAN Notices , Proceedings of the conference on Object-oriented programming, systems, languages, and applications October 1998  
Volume 33 Issue 10
- 2** Java resources for computer science instruction 84%

Joseph Bergin , Thomas L. Naps , Constance G. Bland , Stephen J. Hartley , Mark A. Holliday , Pamela B. Lawhead , John Lewis , Myles F. McNally , Christopher H. Nevison , Cheng Ng , George J. Pothering , Tommi Teräsvirta  
Working Group reports of the 3rd annual SIGCSE/SIGCUE ITiCSE conference on Integrating technology into computer science education December 1998
- 3** Java resources for computer science instruction 84%

Joseph Bergin , Thomas L. Naps , Constance G. Bland , Stephen J. Hartley , Mark A. Holliday , Pamela B. Lawhead , John Lewis , Myles F.

McNally , Christopher H. Nevison , Cheng Ng , George J. Pothering ,  
Tommi Teräsvirta  
ACM SIGCSE Bulletin December 1998  
Volume 30 Issue 4

The goal of this working group was to collect, evaluate, and foster the development of resources to serve as components of both new and revised traditional courses that emphasize object-oriented software development using Java. These courses could, for example, integrate Internet-based distributed programming, concurrency, database programming, graphics and visualization, human interface design and object-oriented development. They could therefore also be suitable as capstone courses in computer ...

**4** Building real-time groupware with GroupKit, a groupware toolkit 84%



Mark Roseman , Saul Greenberg

ACM Transactions on Computer-Human Interaction (TOCHI) March 1996

Volume 3 Issue 1

This article presents an overview of GroupKit, a groupware toolkit that lets developers build applications for synchronous and distributed computer-based conferencing. GroupKit was constructed from our belief that programming groupware should be only slightly harder than building functionally similar single-user systems. We have been able to significantly reduce the implementation complexity of groupware through the key features that comprise GroupKit. A runtime infrastructure

**5** DistView: support for building efficient collaborative applications 84%  
using replicated objects














Atul Prakash , Hyong Sop Shim

Proceedings of the 1994 ACM conference on Computer supported cooperative work October 1994

The ability to share synchronized views of interactions with an application is critical to supporting synchronous collaboration. This paper suggests a simple synchronous collaboration paradigm in which the sharing of the views of user/application interactions occurs at the window level within a multi-user, multi-window application. The paradigm is incorporated in a toolkit, DistView, that allows some of the application windows to be shared at a fine-level of granularity, while still allowin ...

**6** Web-CCAT: a collaborative learning environment for 82%

-  geographically distributed information technology students and working professionals  
Donna Dufner , Ojoung Kwon , Rassule Hadidi  
Communications of the AIS March 1999
- 7** A collaborative fuzzy expert system for the Web 82%  
 Tod A. Sedbrook  
ACM SIGMIS Database June 1998  
Volume 29 Issue 3  
A convergence of Internet and fuzzy logic technologies provides an opportunity for experts and end users to collaborate in developing, refining, and testing knowledge-based systems. Internet technology removes geographical and time-based restraints, and fuzzy rule bases are easier to understand and maintain. This paper describes an architecture and a prototype for developing, delivering, and maintaining expert systems on the World Wide Web. The system's collaboration components allowed experts to ...
- 8** VIRTUS: a collaborative multi-user platform 82%  
 Kurt Saar  
Proceedings of the fourth symposium on Virtual reality modeling language February 1999
- 9** The Upper Atmospheric Research Collaboratory (UARC) 82%  
 Gary M. Olson , Daniel E. Atkins , Robert Clauer , Thomas A. Finholt , Farnam Jahanian , Timothy L. Killeen , Atul Prakash , Terry Weymouth  
interactions May 1998  
Volume 5 Issue 3
- 10** Collaborative virtual workspace 82%  
 Peter J. Spellman , Jane N. Mosier , Lucy M. Deus , Jay A. Carlson  
Proceedings of the international ACM SIGGROUP conference on Supporting group work : the integration challenge: the integration challenge November 1997
- 11** Artefact: a framework for low-overhead Web-based collaborative systems 82%  
 Jeff Brandenburg , Boyce Byerly , Tom Dobridge , Jinkun Lin , Dharmaraja Rajan , Timothy Roscoe  
Proceedings of the 1998 ACM conference on Computer supported cooperative work November 1998

- 12** Teaching multiple programming paradigms: a proposal for a paradigm general pseudocode 80%  
 M. B. Wells , B. L. Kurtz  
ACM SIGCSE Bulletin , Proceedings of the twentieth SIGCSE technical symposium on Computer science education February 1989  
Volume 21 Issue 1  
Initial overexposure to the imperative programming paradigm can make it very difficult to introduce students to other paradigms, such as the functional, object oriented and logical paradigms. It is important that students be exposed to several programming paradigms early. Two techniques commonly used to accomplish this are a "survey of languages" approach and use of a language, such as Scheme, that overlaps several paradigms. We propose the use of a paradigm-general pseudocode t ...
- 13** ICSE '99 conference daily newspaper: Window on the World 80%  
 ACM SIGSOFT Software Engineering Notes July 1999  
Volume 24 Issue 4
- 14** An interview with Peter Donnelley and John Scholes 80%  
 Ray Polivka  
ACM SIGAPL APL Quote Quad March 1998  
Volume 28 Issue 3
- 15** New ideas for generic components in Ada 80%  
 Richard Riehle  
ACM SIGAda Ada Letters September 1998  
Volume XVIII Issue 5  
The creation of reusable software components is an important part of modern software practice. Generic templates are one technique for designing these components. A generic template is a module containing algorithms which can operate on some class of data types where the specific data type is not known until later in the development process. Many languages, including Ada, support this technique. In Ada, generic templates must be type-safe at compile time. We examine some features of Ada which al ...
- 16** Computer-mediated communication in collaborative educational settings: report of the ITiCSE '97 working group on CMC in collaborative educational settings 80%  
 Ursula Wolz , Jacob Palme , Penny Anderson , Zhi Chen , James Dunne , Göran Karlsson , Atika Laribi , Sirkku Männikkö , Robert Spielvogel , Henry Walker  
ACM SIGCUE Outlook October 1997

## Volume 25 Issue 4

- 17** Computer-mediated communication in collaborative educational settings (report of the ITiCSE '97 working group on CMC in collaborative educational settings) 80%



Ursula Wolz , Jacob Palme , Penny Anderson , Zhi Chen , James Dunne , Göran Karlsson , Atika Laribi , Sirkku Männikkö , Robert Spielvogel , Henry Walker

The supplemental proceedings of the conference on Integrating technology into computer science education: working group reports and supplemental proceedings June 1997

- 18** Assessing process-centered software engineering environments 80%



Vincenzo Ambriola , Reidar Conradi , Alfonso Fuggetta

ACM Transactions on Software Engineering and Methodology (TOSEM) July 1997

Volume 6 Issue 3

Process-centered software engineering environments (PSEEs) are the most recent generation of environments supporting software development activities. They exploit an representation of the process (called the process model that specifies how to carry out software development activities, the roles and tasks of software developers, and how to use and control software development tools. A process model is therefore a vehicle to better understand and communicate the process. If ...

- 19** An architecture for WWW-based hypercode environments 80%



Gail E. Kaiser , Stephen E. Dossick , Wenyu Jiang , Jack Jingshuang Yang

Proceedings of the 19th international conference on Software engineering May 1997

- 20** Personal distributed computing: the Alto and Ethernet software 80%



Butler Lampson

Proceedings of the ACM Conference on The history of personal workstations January 1986

The personal distributed computing system based on the Alto and the Ethernet was a major effort to make computers help people to think and communicate. The paper describes the complex and diverse collection of software that was built to pursue this goal, ranging from operating systems, programming environments, and communications software to printing and file servers, user interfaces, and applications such as editors, illustrators, and mail systems.

---

**Results 1 - 20 of 102**    **short listing**Prev  
Page**1****2****3****4****5****6**Next  
Page

---

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2002 ACM, Inc.